

Shell & Les Commandes de base

Syntaxe	Description
Utilitaires d'aide	
man [Commande]	Permet de fournir une page manuelle d'une commande envoyée en argument.
info[commande]	Permet de fournir une page info d'une commande envoyée en argument.
apropos[Commande]	Permet de lister les manuels dont la description comprend les mots passés en arguments.
Manipulation des répertoires (Change Directory)	
cd /chemin repertoire	Déplacement dans l'arborescence des répertoires vers le repertoire spécifié par le chemin donné à cd.
cd .	Positionnement sur le repertoire courant.
cd ..	Positionnement sur le repertoire parent du repertoire courant
cd ~	Positionnement sur le repertoire personnel de l'utilisateur connecté
cd /	Permet de remonter à la racine de l'ensemble du système de fichiers.
pwd	Positionnement courant.
Variable d'environnement	
echo \$PATH	Affiche les répertoires spécifiés dans la variable PATH sur votre système.
echo \$HOME	Afficher le chemin d'accès absolu vers le repertoire personnel de l'utilisateur actuel.
echo \$ SHELL	Indique l'interpréteur shell utilisé par défaut.
echo \$ LOGNAME	Pour le nom de l'utilisateur
echo \$ HISTFILE	Pour le fichier historique.
echo \$ HISTSIZE	Pour la limite de commandes historiques accessibles.
env	Pour afficher toutes les variables d'environnement
Lister les le contenu d'un repertoire (Liste Sorted)	
ls chemin_repertoire	Lister le contenu d'un repertoire.
ls -a chemin_repertoire	Lister le contenu du répertoire y compris les fichiers cachés.
ls -l chemin_repertoire	Lister le contenu d'un repertoire de manière détaillée
ls -d chemin_repertoire	Afficher les répertoires sans lister leurs contenus
ls -i chemin_repertoire	Afficher le numéro d'inode.
Créer répertoires : (make directory)	
mkdir chemin_repertoire	Créer un repertoire.

mkdir -p rep1/rep2/rep3	Créer un répertoire et ses sous répertoires associés. (Créer toute une branche de l'arborescence)
Créer fichier (make_file)	
touch nouveau_fichier	Création d'un nouveau fichier
nano fichier	Lancer un éditeur de texte en mode texte plein écran pour éditer le fichier
Gedit fichier	Editer un fichier.
Copier fichiers / répertoires (copy)	
cp chemin_fichier_source chemin_fichier_destination	Copier le fichier source dans le chemin destination.
cp -i source destination	Avertir de l'existence d'un fichier du même nom et demande s'il peut ou non le remplacer.
cp -r source destination	Copier de manière récursive l'ensemble d'un répertoire et de ses sous répertoires.
Cp -p source destination	Permet de copier les fichiers tout en préservant les attributs spécifiques tels que les permissions, l'heure de modification, l'heure de création, l'heure d'accès, les propriétaires et les groupes.
Supprimer des fichiers / répertoires : (remove & remove directory)	
rm chemin_fichier	Supprimer un fichier.
rm -r chemin_fichier	Supprimer un répertoire et ses sous répertoires demande s'il peut ou non le remplacer.
rm -i chemin_fichier	Demander une confirmation avant la suppression du fichier.
rmdir chemin_repertoire	Supprimer un répertoire vide
rm -f	Permet de supprimer les fichiers et répertoires sans demander de confirmation à l'utilisateur
Déplacer ou renommer un fichier: (move)	
mv fichier1 fichier2	Renommer fichier1 en fichier2.
mv fichier1 chemin	Déplacer le fichier
mv -i fichier1 fichier2	Demande pour chaque fichier et chaque répertoire s'il peut ou non le déplacer.
Créer un lien (physique ou symbolique): (link)	
ln fichier hardlink	Créer un lien physique.
ln -s fichier symboliclink	Créer un lien "symbolique" (raccourci).
GREP: Globally search for Regular Expression and print it	
grep chaîne fichier	Afficher toutes les lignes contenant « chaîne » du fichier.
grep -r chaîne rep	Rechercher d'une façon récursive dans tous les fichiers et sous-dossiers du rep.
grep -i chaîne fichier	Ignorer la différence entre majuscule et minuscule.
grep -v chaîne fichier	Afficher les lignes ne contenant pas la chaîne.
grep -n chaîne fichier	Chaque ligne contenant la chaîne est numérotée.
grep -c chaîne fichier	Compter le numéro d'apparence de la chaîne
Recherche des fichiers (find, locate)	
locate fichier	Rechercher ou localiser le fichier en donnant son chemin
find repertoire option	Rechercher des fichiers ou des répertoires dans «répertoire » en utilisant plusieurs options de recherche

find -name	Recherche par nom de fichier.
find -type	Recherche par type de fichier.
Find -user	Recherche par propriétaire.
Find -group	Recherche par appartenance à un groupe.
Find -size	Recherche par taille de fichier.
Find -atime	Recherche par date de dernier accès.
Find -mtime	Recherche par date de dernière modification.
Find -ctime	Recherche par date de création.
Find -perm 777	Recherche par autorisations d'accès.
Find -links	Recherche par nombre de liens au fichier.
WC: Word Count	
wc fichier	Affiche le nombre de lignes, nombre de mots et le nombre d'octets du fichier.
wc -l fichier	Affiche le nombre de lignes d'un fichier.
wc -w fichier	Affiche le nombre de mots d'un fichier.
wc -c fichier	Affiche le nombre d'octets dans un fichier.
wc -m fichier	Affiche le nombre de caractères d'un fichier.
SORT Trier	
sort fichier	Trier les lignes d'un fichier par défaut (ordre alphabétique de la première chaîne au début de la ligne). A -> Z
sort -n fichier	Trier par ordre numérique.
sort -r fichier	Trier inverse. Z -> A
sort -t [séparateur] -k[n]	Trier la «n»ème colonne selon le séparateur
uniq fichier / sort -u	Supprimer les doublons (applicable que sur les fichiers triés)
CAT : Concaténation	
cat fichier	Afficher le contenu d'un fichier
cat fichier1, fichier2	Afficher le contenu de 2 fichiers
cat -n fichier	Afficher le contenu du fichier avec des numéros de ligne. (en prenant en compte les lignes vides)
cat fichier sed '/^\$/d'	sed '/^\$/d' permet de faire l'affichage sans prendre en compte les lignes vides dans le fichier
HEAD / TAIL	
head fichier	Affiche par défaut les dix premières lignes d'un fichier
head -n fichier	Affiche les n premières lignes du fichier.
tail fichier	Affiche par défaut les dix dernières lignes d'un fichier
tail -n fichier	Affiche les n dernières lignes du fichier.
CUT	
cut fichier	Permet de filtrer (supprimer) une partie de chacune des lignes d'un fichier.
cut -d[délimiteur] fichier	L'option -d permet d'exprimer le caractère séparateur de champ [délimiteur]. Le caractère séparateur par défaut est la tabulation.
cut -f fichier	Permet à définir <nb> champ à garder
Gestion des droits	
chmod options modes fichier	Changer les permissions d'accès d'un fichier ou d'un répertoire.

<code>chmod [u g o [+ - =] [r w x]</code> <code>nom_du_fichier</code> <code>chmod -R [u g o] [+ - =] [r w x]</code> <code>r = 4 / w = 2 / x = 1</code> <code>nom_du_répertoire</code> <code>chmod [valeur *] [nom du fichier]</code>	
Redirection	
<code>Cmd (terminale) > fichier</code>	Si le file existe, alors il est écrasé et recréé. Si non il le crée et redirige la sortie de cmd vers ce fichier
<code>Cmd >> fichier</code>	Rajoute le sorite de cmd au fichier
<code>Cmd 2 >> fichier / cmd 2 >fichier</code>	De même principe mais cette fois il redirige le flux d'erreur vers le fichier

Environnement Graphique : Prompt

❖ Le prompt est structuré comme suit :

```
|esprit@esprit-virtual-machine:~$
```

1
2
3
4

- ❶ Le login ou le nom de l'utilisateur connecté
- ❷ Nom de l'hôte ou de la machine
- ❸ La position actuelle dans le système de fichiers
- ❹ Types d'utilisateurs (\$: utilisateur simple / # : admin)

Terminal= Un environnement graphique d'entrée / sortie de texte qui permet à l'utilisateur d'introduire des commandes Shell.

Shell= Un interpréteur de commande + Un langage de programmation.

Shell présente une interface entre l'utilisateur et le système d'exploitation.

Ainsi, le shell est un programme qui permet à l'utilisateur de dialoguer avec le cœur du système (le noyau).

Shell : Interpréteur de commandes	Shell : Langage de programmation
<ul style="list-style-type: none"> ❖ Shell interprète des commandes, des programmes ou des fichiers exécutables. ❖ Il assure : <ul style="list-style-type: none"> • Affichage de l'invite de commande ou prompt (\$) d'attente de lecture au clavier. • Lecture d'une commande (validée par Return ou Entrée). • Analyse syntaxique (découpage en mot). • Interprétation des caractères spéciaux. • Exécution de la commande et retour au début. 	<ul style="list-style-type: none"> ❖ Shell fournit à l'utilisateur un langage de programmation interprété. ❖ Il comporte : Les notions de variable, d'opérateur arithmétique, de structure de contrôle, de fonction, des opérateurs spécifiques (;)

- Chaque utilisateur possède un shell par défaut, qui sera lancé à l'ouverture d'une invite de commande.
- Deux modes d'utilisation du Shell :
- **Interactif** : l'utilisateur saisit et exécute ses lignes de commandes une par une, dans un terminal ;
- **Non interactif** : Shell interprète un ensemble de commandes à partir d'un fichier appelé shell script.

Les commandes Shell :

date : affiche la date, [jour de la semaine, mois, jour, temps, timezone, année]

cal : affiche le calendrier

who : affiche les utilisateurs connectés

whoami : affiche votre nom d'utilisateur

uname -a : affiche toutes les informations sur le système

whatis : affiche une description de la commande

hostname: afficher le nom de la machine

halt ou shutdown -h now ou init 0: éteindre l'ordinateur

reboot ou shutdown -r now ou init 6: rebooter l'ordinateur

lsb_release -d : afficher le nom de la distribution.

Inode : «Index» Node

Les inodes (contraction de «index» et «node», en français : nœud d'index) **sont des structures de données contenant des informations concernant les fichiers stockés** dans certains systèmes de fichiers (notamment de type Linux/Unix).

- ❖ À chaque fichier correspond un numéro d'inode(i-number) dans le système de fichiers dans lequel il réside, unique au périphérique sur lequel il est situé.
- ❖ Les inodes sont créés lors de la création du système de fichiers.
- ❖ La quantité d'inodes (généralement déterminée lors du formatage et dépendant de la taille de la partition) indique le nombre maximum de fichiers que le système de fichiers peut contenir.

Lien Physique Vs Lien Symbolique :

Un lien est un fichier spécial qui permet de faire plusieurs références à un même fichier sur le disque dur. L'intérêt est de pouvoir accéder à un fichier de plusieurs endroits de l'arborescence. Il existe deux types de liens : les liens physiques et les liens symboliques.

Lien physique [Commande Ln]

Le système de fichier de Linux enregistre les fichiers sur la base d'un numéro et pas sur la base d'un nom. Grâce à un lien physique, un fichier peut avoir **plusieurs noms/ chemin d'accès**, à un même fichier en pointant sur un numéro unique d'inode qui identifie le fichier dans le système de fichiers.

Lien Symbolique Commande Ln -s

Les liens symboliques ressemblent plus aux « **raccourcis** » sous Windows. La plupart du temps, on crée des liens symboliques sous Linux pour faire un raccourci, et non des liens physiques.

Le principe du lien symbolique est que l'on crée un lien vers un autre nom de fichier. Cette fois, on pointe vers le nom de fichier et non vers l'inode directement

- ❖ Un lien physique utilisant **le même numéro d'inode**, il se limite à la même partition.
- ❖ Pour traverser les limites des partitions, Unix introduit des liens dits symboliques.
- ❖ Un lien physique pointe sur la même inode du fichier contrairement au lien symbolique.
- ❖ **Un lien physique** ne peut être créé que pour **des fichiers**.
- ❖ Un lien symbolique peut être créé pour un fichier comme pour un répertoire (Dossier).

Exercice 1 :

1. Créer un dossier nommé « catalogue » sous votre répertoire personnel.

mkdir catalogue

2. Créer sous « catalogue » deux fichiers nommés « influencer.csv » et « lifestyle.txt »

cd catalogue

touch influencer.csv lifestyle.txt

3. Editer le fichier « influencer.csv », copier le contenu suivant, enregistrer et quitter :

nano influencer.csv / ou bien gedit influencer.csv

Nom, Nationalité, Expérience, Catégorie, Adresse email
Fatma Bououn ,Tunisienne, 3 ans , Lifestyle, fatma.b@faza.tn
Fatma Bartakis , Tunisienne, 5 ans , Lifestyle , fatma.bartakis@gmail.com
Nidhal Saadii, Tunisien , 4 ans , Comedy, nidhal.saadi@gmail.com
Marwa Agerbi, Tunisienne, 10 ans , Beauty, contact@marwaagerbi.com
Sami Chaafi , Tunisien, 3 ans , Content Creation, samichaafi@gmail.com
Mourad Rouge , Tunisien, 6 ans , Comedy , mourad_labidi@hotmail.com

4. Afficher le contenu de votre fichier « influencer.csv » sur la console.

Cat influencer.csv

5. Afficher seulement les 3 premières et les 2 dernières lignes du fichier « influencer.csv » sur la console.

head -3 influencer.csv && tail -2 influencer.csv

6. Créer un deuxième dossier sous votre répertoire personnel qui s'intitule « classement ».

cd ..

mkdir classement

7. Déplacer le fichier « influencer.csv » vers « classement » et le renommer « copieinf.csv ».

cp -p \$HOME/catalogue/influencer.csv \$HOME/classement/copieinf.csv

8. **Extraire** les noms et les adresses mails des influencer appartenant à la catégorie **LifeStyle** du fichier « copieinf.csv » et les enregistrer dans « Lifestyle.txt ».

grep -i lifestyle copieinf.csv | cut -d ',' -f 1,5 > \$HOME/catalogue/Lifestyle.txt

9. Chercher dans les fichiers du répertoire « catalogue », l'influenceuse dont le nom est « Marwa Agerbi ». **grep -ri "Marwa Agerbi" catalogue**

10. Trier les lignes du fichier « influencer.csv » par ordre alphabétique et les mettre dans un nouveau fichier « tri.txt » et afficher le nombre des influencer.

cd catalogue

sort influencer.csv > tri.txt

wc -l tri.txt

11. Trier les lignes du fichier « influencer.csv » suivant l'ancienneté des influencer plus ancien vers la moins ancien et les placer dans « tri_exp.txt ».

sort -t ',' -k3nr influencer.csv > tri_exp.txt

Correction :

```
1- esprit@esprit-virtual-machine:~$ mkdir catalogue
esprit@esprit-virtual-machine:~$ ls -l
2- esprit@esprit-virtual-machine:~$ cd catalogue
esprit@esprit-virtual-machine:~/catalogue$ touch influencer.csv lifestyle.txt
3- esprit@esprit-virtual-machine:~/catalogue$ nano influencer.csv
4- esprit@esprit-virtual-machine:~/catalogue$ cat influencer.csv
5- esprit@esprit-virtual-machine:~/catalogue$ head -3 influencer.csv && tail -2
influencer.csv
6- esprit@esprit-virtual-machine:~$ cp $HOME/catalogue/influencer.csv $HOME
/classement/copieinf.csv
7- esprit@esprit-virtual-machine:~/classement$ grep -i Lifestyle copieimpf.csv | cut -d ',' -f
1,5 > $HOME /catalogue/lifestyle.txt
8- esprit@esprit-virtual-machine:~$ grep -ri "Marwa Agerbi" catalogue
9- esprit@esprit-virtual-machine:~/catalogue$ sort influencer.csv > tri.txt
esprit@esprit-virtual-machine:~/catalogue$ wc -l tri.txt
10- esprit@esprit-virtual-machine:~/catalogue$ sort -t ',' -k3nr influencer.csv (ordre
numérique décroissant
esprit@esprit-virtual-machine:~/catalogue$ sort -t ',' -k3n influencer.csv (ordre numérique
croissant
```

Exercice 2 sur la commande find :

1- Comment chercher tous les fichiers dont les noms commencent par un «A» majuscule ou une minuscule, suivi d'éventuellement quelques lettres ou chiffres, et se terminent par un chiffre entre 3 et 6 ?

2- Comment fait-on pour indiquer que le fichier recherché a été modifié il y a plus de 30 jours ? Il y a 30 jours ? Il y a moins de 30 jours ?

3- Rechercher tous les dossiers qui se trouvent sous /etc d'extension .d

4- Chercher tous les fichiers dans /usr dont la taille est supérieure à 5Mo

5- Afficher toutes les lignes sous /etc qui ne contiennent pas le mot "protocols"

Correction :

1- find . -name '[aA]*[3-6]'

2-

- -mtime 30 : le fichier a été modifié il y a 30 jours;
- -mtime +30 : le fichier a été modifié il y a 30 jours ou plus;
- -mtime -30 : le fichier a été modifié il y a 30 jours ou moins.

3- find /etc -type d -name "*.d"

4- find . -type f -size +30k

find /usr -type f -size +5M

5- grep -v "protocols" /etc/*

Exercice 3 sur la commande grep:

Grep ^emna fichier1 je cherche le nom emna au début de la ligne dans le fichier1

Grep \$emna fichier1 je cherche le nom emna à la fin de la ligne dans le fichier1

1- Chercher toutes les lignes commençant par «a» ou «A»

2- Chercher toutes les lignes finissant par «rs»

3- Chercher toutes les lignes contenant au moins un chiffre

4- Chercher toutes les lignes commençant par une majuscule

5- Chercher toutes les lignes commençant par «B», «E» ou «Q»

6- si on cherche tous les «a», sauf ceux suivis de «b», «c» ou «t», on écrit

Correction :

1- grep -i '^a' fichier ou grep '^aA' fichier

2- grep 'rs\$' fichier

Le signe "\$" est utilisé pour spécifier que la recherche doit être effectuée à la fin de la ligne.

Le symbole "^" est utilisé pour indiquer que la recherche doit être effectuée au début de la ligne

3- grep '[0-9]' fichier

4- grep '^[A-Z]' fichier

5- `grep '^[BEQ]' fichier`

6-

`grep 'a[^bct]'`

`grep -v '[.,;?!]$\'` fichier

l'option -v, qui prend toutes les lignes où ne figure pas une chaîne de caractères donnée